P-14-316-US

VERIFICATION OF TRANSLATION

To whom it may concern,

I,

Mr. Joel Wenger, at Leman Consulting SA, route de Clementy 63, 1260 NYON,

hereby certify that to the best of my knowledge and belief the following is a true translation compared by me and for which I accept responsibility of the following attached document :

International Application No. PCT/IB00/01301 as filed on September 14th, 2000.

" METHOD AND SYSTEM FOR TRANSMITTING A CHAIN OF MESSAGES FOR DATABASE "

Dated this ..... 4th ..... day of ..... October 2005

Signature .....

MESSAGE CHAINING TRANSMISSION PROCESS AND SYSTEM FOR DATA BASES.

Introduction

The present invention concerns a process and system of database updating, and in particular during the transmission of a chain of messages.

Background of the Invention

In a system comprising a management centre and a plurality of subscribers over a wide territory, the sending by telephone or hertz route of updating information for the database of these subscribers is known. These messages are addressed, either to all subscribers, or to one subscriber in particular, that is to say that it contains a subscriber module address.

These messages are for the administration of the system and are themselves superposed to the useful data such as video, audio or data. One understands that the place used by these messages is limited. Another limitation applies to the length of the message which is limited by the fact that the useful data can be interrupted only for a short moment. One understands that, in the example of an audio/video transmission, the emission channel can be interrupted only for a short moment so that no visual impact will be perceptible.

This is why, for transmission of a large amount of data, it was necessary to divide them in a large number of messages.

These messages are sent in sequence on the network, in a logic order, that is to say one after the other, separated by a short interval, for example one second.

As certain systems of this type do not use return channel towards the managing centre, as for example a modem, it is difficult for the managing centre to know if the data sent is arrived correctly. In this way, it is obliged to repeat these messages periodically so as to statistically ensure that each message is arrived at destination.

A subscriber module includes schematically a digital receiver, either audio, or video or data, and possibly a combination of these three types, a decoder able to extract the management messages, these latter being directed towards a security module comprising the subscriber database. This security module can be directly installed in the subscriber module or, for security and cost reasons, it can be as a detachable module such as an smart card or microchip card.

The messages arriving to the security module are processed by the command interpreter. It is possible that the messages do not arrive in the broadcasting order because of interferences in the transmission or simply as the subscriber's unit was not switched on at the moment of the sending of previous messages. It is necessary to specify that prior each processing, each message is first decrypted and controlled for its authenticity. A message which does not satisfy the control criteria is rejected. In this case, the security module will receive for example the third message before the first and second messages. The execution of the third message without the prerequisite execution of two previous messages can lead to blocking of the database or to generate an error.

A first solution consists in memorising all messages constituting a chain and, when it is complete, to go on to its processing. This solution has the drawback to set the length of the maximum chain according to the memory available.

The memory capacity of detachable smart cards is limited, which obliges the card to process each message when they arrive.

Brief Summary of the Invention

The problem that the present invention proposes to solve, is to suppress on the subscriber's database the disastrous effects due to the execution of messages in an order different to that initially foreseen.

This aim is fully reached by a transmission process of a chain of subscriber's database management messages, this process consisting of associating a conditional block which determines if the message is to be processed with reference to all or part of others element of the chain and the conditions bound to the previous processing of all or part of other elements of the chain.

In fact, due to this new conditional block included in each message elements of a chain, it is possible to determine if this message can be processed separately or must satisfy the conditions of processing of the messages supposed to be received previously. It is obvious that this test allows also the determination if the current message evaluation has already been processed.

To reach this aim, the security module disposes of a memory organized under the form of table indicating, for each chain, which are the messages member of this chain that have already been processed.. After processing of all elements of the chain, the table of this chain is maintained in order to avoid that the resending of the same chain restarts its execution. It can be deleted on request by the management centre or after a predefined time.

The conditional block contained in the message does not contain only a simple indication binding the processing of the current message to condition of having carried out the execution of previous messages, but also covers more complex functions, such as conditions related to each element of the message chain. For example, it is possible to define the processing of element 4 of the chain on the condition that either element 1 or 2 is processed and that element 3 is imperatively processed. We will thus have the function:

F(4) = (1 or 2) and 3.

We take the example of the arrival to the security module of a message member of family 5, this message being the element 4 of this family. The first operation will be to determine if its processing is subject to other conditions. If this is not the case, it can be processed immediately. It should be noted that to chain messages does not mean that the processing must be made in the index order of the chain. One can imagine the case where one loads a bulky software, and for this reason, one divides it to transmit it in a chain of messages. Each of these messages contains a loading address and the corresponding data. This is why an element of the chain can be processed in a indifferent order. On the other hand, the last element of the chain setting up this new software will contain a condition stating that all elements of the chain must have been carried out in order that this software can be executed. When

this condition has been satisfied, the table correspondent to this family indicates that all messages have been carried out.

According to a variant of the invention, the conditional block is divided in two parts, the one called "operation" to describe the type of logic function and the other called "related element" to describe on which other elements the operation must apply. The format of the part "related element" corresponds to the format used in the table stored in the database designating the state of processing the elements of the chain. In this way, the logic comparison is greatly facilitated.

According to other embodiments, the conditional block refers not to all the other elements of the chain, but to some only. It would be for example possible to referr to three previous elements and not to all the elements. This allows the reduction of the length of the conditional block and takes into account the fact that an interference rarely exceeds the time of three messages. According to another example, one could define a chain structure where only the last element contains a conditional block.

This structure allows, unlike the solutions of the prior art, to reject only a minimum of messages. In fact, when a message is missed in a chain, all the following messages were rejected until the new passage of the missing message. The execution of a chain was in this way dependent upon the continuous reception of elements of the chain, each element missing leading to the rejection of all messages having an higher index than the missing message.

According to an embodiment of the invention, the subscriber module, besides sending the messages to the security module, includes a memory to memorise them as soon as they arrive.

Therefore, it is possible that the absence of a message containing a condition on a preceding message leads to reject all the following messages. When this awaited message arrives, it is of course processed authorising the processing of other messages. It is possible otherwise that a long time elapsed before these missing messages are present in the transmission with the risk that some are rejected, for example due to the bad quality of the connection between the managing centre and the subscriber module.

To minimize the number of repeated messages necessary for the completion of the chain, the security module can accede to the memory located in the subscriber module since it contains all the messages in their arrival order. Thus, as soon as the missing message arrives and its processing completed, the security module asks the reading of the memory to process all the messages which have been rejected because of the condition on the missing message.

An important aspect of the invention lies in presenting each message to the security module while storing it in memory in the subscriber module. This principle can include exceptions when some messages are not destined to the security module but only to the module of the subscriber. Thus, even if some messages are rejected by the security module as the conditions are not fulfilled, this system knows that this message is contained in the memory of the subscriber module and can, when the condition is fulfilled, accede to this memory to proceed these messages instead of awaiting a next passage of following messages.

In an embodiment, the memory of the subscriber module is organised as a stack with entry in series, each new entry causing the displacement of previous entry (first-in first-out).

The reading by the security module can be realised in different ways. It can ask the transmission of an exact address of the memory. Nevertheless, an important aspect of the security in this kind of application lies in the confidentiality of the organisation of the data. For this reason, instead asking the transmission of a specific address, the security module asks the subscriber module to submit all or part of messages contained in its memory. It is the task of the security module to sort out between the messages already carried out and the messages to carry out.

Brief description of the drawings

The invention will be better understood based on the following detailed description which refers to annexed drawings which are given by way of a non limitative example, wherein:

- Figure 1 represents a message sent according to the systems of the prior art;
- Figure 2 represents a message sent according to the invention;

- Figure 3 represents one embodiment for updating the temporary memory of the subscriber module.

Detailed description of the invention

In Figure 1 the different blocks of a message which take part in the function of chaining are represented schematically. We find a first header block HD, which describes the kind of message, and contains the information that this message is part of a chain. To form the chain, a second family block FM indicates to which family this message belongs. In fact, it is possible that several chains are transmitted simultaneously and in this case the identification of the family is necessary. Now that the family is defined, the subsequent block FI is used to identify each element of the family and its place in the chain. So, with these two data, each element of the family can be placed at the right place in the chain. It is known to indicate in one or the other of control blocks FI or FM the maximum number of the element of the family. This function can equally be obtained by a particular marking of the last element of the family.

In the example of figure 2, in the message of Figure 1, showing the two blocks FM and FI, one add a supplementary block CD which determines a condition to carry out this message. According to a first embodiment of the invention, this block is constituted by a bit which indicates if the previous message should or should not have been executed. If this condition is requested, the interpreter in charge of the operations on the database, will verify if the previous message has been executed properly and in the positive event, will execute this new message.

In another embodiment, this conditional block CD is constituted by a field comprising groups, a group for each element of the chain. Each group contains a condition on an element of the chain and can have several meanings, for example the condition "must have been executed", "can be executed" or "must not be executed". The latter condition generally is the complement of the first.

We take the example of a chain of 6 elements, the element 3 should imperatively be carried out before element 5. In this case, one can specify in message 3 that it should not be carried out if message 5 was not processed. This condition can lead to a locking if one does not specify the inverted condition in message 5. In this case,

message 5 will contain the condition "must be executed", in reference to the message 3 in order that if message 5 arrived before 3, it will be not processed.

In Figure 3, an implementation of the memory M of the subscriber module and the connection with the security module are represented. The incoming flux is firstly filtered by a module SEL, which has the task to separate the managing messages from other data. These messages are then transmitted to the selection module SW which has the task to send them to different modules i.e. the security module SM, to the processing centre CTR of the subscriber module STB and to the memory M of the subscriber module. The storage in memory of these messages causes the increment of the input message pointer so that no message will be lost, the oldest message being then eliminated from the memory. In the same way, these messages are transmitted to the security module, represented here as an smart card SM. This card SM contains a first memory managing module GM and a control interpreter INT for managing the controls of the database BD. This memory manager GM can dialogue with the processing centre CTR by the connection I/O and by this means, to influence the connections in the selection module SW. The dotted line represented in Figure 3 represents the subscriber module STB. All the managing messages addressed to the security module SM are directed by the selector SW to the security module, in particular to the memory manager GM, then are transmitted to the control interpreter if the processing conditions are fulfilled. The memory manager GM updates the table of messages processes to make the necessary comparisons at the moment of the arrival of a new message. The connection with the smart card SM is of in/out type and in this way information and controls can be sent at destination of the subscriber module, this connection being represented by the line I/O. As explained previously, the memory M is physically in the subscriber unit STB. This is why the card SM can, by the intermediate of the line I/O, ask the availability of a memory section so as to be able to store the messages of a chain. In our example, the maximum number of elements in a chain does not exceed 16. So, at the arrival of the first element of the chain, the card SM, by the line I/O, requests the reservation of at least 16 memory places. If, during the transmission of this first chain, another chain is announced, the card will ask the reservation of 16 new places in order to ensure the storage of a maximum number of the chain according to the receiving conditions.

In order to read the data contained in the memory M, for example the position M3, the card SM can order, through the selection module SW, the address multiplexer AMUX to return the content of this memory position. In order to forward these data towards the card, a data multiplexer DMUX has the function to read the memory position required and to transfer it towards the card. These different transfers are directed by the selection module SW.

When the processing of the chain has been interrupted due to a interference on a message for example, the other messages continue to be stored in the memory of the subscriber module. When the missing message is retransmitted by the managing centre, it is executed properly and the memory manager GM recalls all the other messages of the chain acceding the memory of the subscriber module. In this case, the entry of the smart card SM is not made any longer on the arrival of messages but on the contents of the memory M. This access to memory M can be made in direct access specifying a memory address, or by sequential access reading the messages in their arrival order.

In an embodiment, the memory M is organized as a memory buffer of a fixed length according to the availability of the free memory of the subscriber module. This memory includes an input pointer increased on each message introduced in the memory, and an output pointer increased on each reading by the memory manager GM.

The communication possibility between the card SM and the subscriber module STB, in particular the centre CTR, authorises more complex functions. One problem frequently met at the moment of the replacement of one or the other of the elements of the system, either the card or the subscriber module, is to ensure the compatibility of functions with the material of previous generations. For this, it is interesting to allow communication between the different elements in order to establish the functions available in each of them; this is the task of the line I/O which allows to send instructions of the card to the subscriber module. These instructions can, for example, ask the subscriber module to communicate its audio, video or data functions, the generation of the module or the software version. To answer to this request, the module STB disposes of means to compose a managing message and to transmit it, in the memory M for further reading by the card, or directly to the card, such as

such as represented in Figure 3.

According to another embodiment of the invention, the module STB disposes of a connection by modem with the managing centre. In this case, the announcement of resources can be made by the module STB to the managing centre through the modem, on request of the security module SM.

As indicated in Figure 3, the module STB receives in the same way the managing messages coming from the managing centre. The messages arriving to the processing centre CTR can contain a configuration request instruction. The response can be made by the modem or be transmitted to the card SM. Some of these managing messages are only destined to the module STB and the processing centre CTR, responsible to the management of the module STB, will not transmit them to the security module SM or to the memory M.